



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/649,903	08/26/2003	Mahesh A. Ramchandani	5150-77400	5752

7590 09/15/2008
Jeffrey C. Hood
Meyertons, Hood, Kivlin,
Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767

EXAMINER

MITCHELL, JASON D

ART UNIT	PAPER NUMBER
----------	--------------

2193

MAIL DATE	DELIVERY MODE
-----------	---------------

09/15/2008

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/649,903	Applicant(s) RAMCHANDANI, MAHESH A.	
	Examiner Jason Mitchell	Art Unit 2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 27 June 2008.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 76-86,88-107 and 109-114 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 76-86,88-107 and 109-114 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--------------------------------------------------------------------------------------|-------------------------------------------------------------------|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

In view of the appeal brief filed on 6/27/08, PROSECUTION IS HEREBY REOPENED.

New grounds of rejection are set forth below.

To avoid abandonment of the application, appellant must exercise one of the following two options:

(1) file a reply under 37 CFR 1.111 (if this Office action is non-final) or a reply under 37 CFR 1.113 (if this Office action is final); or,

(2) initiate a new appeal by filing a notice of appeal under 37 CFR 41.31 followed by an appeal brief under 37 CFR 41.37. The previously paid notice of appeal fee and appeal brief fee can be applied to the new appeal. If, however, the appeal fees set forth in 37 CFR 41.20 have been increased since they were previously paid, then appellant must pay the difference between the increased fees and the amount previously paid.

A Supervisory Patent Examiner (SPE) has approved of reopening prosecution by signing below:

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193

DETAILED ACTION

1. This action is in response to an appeal brief filed on 6/27/08.
2. Claims 76-86, 88-107 and 109-114 are pending in this case.

Response to Arguments

Independent Claim 76

3. In the 2nd full par. on pg. 11, the applicant states:

However, [col. 4, lines 4-5] of Grey has nothing to do with a control that includes pre-existing first functionality for determining the steps in the test executive sequence. The cited portion of Grey instead relates to step types. A step type "essentially comprises a custom set of properties and/or operations associated with a step." The user can define various step types. When a given step is created and added to the sequence, the user can request that the step be of a particular step type. Grey teaches that, "For each type that a file uses, the TestStand system stores the definition of the type in the file." (Col. 3, lines 38-40). The teaching cited by the Examiner refers to the file subsequently being loaded after the step type definitions have been stored in it. Grey teaches here that, "In response to the user loading the file, the TestStand Engine automatically determines the type being loaded, and then automatically determines if the loaded type conflicts with one or more previously loaded/registered types." (Col. 4, lines 4-5). Thus, the cited passage teaches nothing whatsoever about a control that includes pre-existing first functionality for determining the steps (not the step types!) in the test executive sequence.

The examiner respectfully disagrees. First it is noted that the applicant has failed to indicate the perceived distinction between the claimed determining a "step" and Grey's determining of a "step type" (see col. 3, lines 35-38). Accordingly, the applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references.

Further, those of ordinary skill in the art would have recognized that determining the 'type' of a step must first include identifying the step whose type is being identified (as well as the "set of properties and/or operations associated with [the] step").

Additionally, Grey discloses "determining if the name of the loaded type conflicts with the name of any of the previously loaded/registered types" (col. 4, lines 9-13). From this it can be seen that Grey's system has determined the "name" of a type and thus provides a control which "determine[s] the steps in the test executive sequence".

Still further, on pg. 1 of the specification as originally filed the applicant defines a step as:

Step – An action that the user can include within a sequence of other actions. A step may call a test module to perform a specific test.

In col. 1, lines 58-60 Grey defines a step as:

Step — Any action, such as calling a test module to perform a specific test, that the user can include within a sequence of other actions.

In view of these definitions those of ordinary skill in the art would have recognized that Grey's 'Steps' are analogous to the applicant's claimed 'steps'.

4. Starting in the first full par. on pg. 12, the applicant states:

The limitations in question relate to the "at least a subset of the steps" being automatically displayed in the GUI element during execution of the run-time operator interface application. However, the cited portions of Grey teach nothing whatsoever about any steps of the sequence being displayed during execution of the run-time operator interface application. ...

Thus, Grey is here [col. 3, lines 16-18] simply describing the architecture of the test executive system, teaching that the sequence editor and the operator interface programs interface to the test executive engine, e.g., as shown in FIG. 2. In contrast, claim 76 recites, "configuring a binding between the GUI element and the control".

Art Unit: 2193

The cited portions of Grey teach nothing about configuring a binding between a GUI element and a control.

The examiner respectfully disagrees. First, Grey's fig. 4 shows an operator interface application (i.e. a GUI element of an application) displaying the steps in a test executive sequence (e.g. "Check for Proper Use", "Clear Report", "PreUUTLoop Callback"). Further, as Grey discloses a GUI element of an operator interface application, the data in that GUI element must necessarily be shown at runtime. In other words, if the application is not running the GUI can not be displayed. Further, Grey discloses the GUI element (see Fig. 4) "interfaces" with the control (col. 4, lines 4-7 "the TestStandEngine"). The limitation in question only broadly recites "configuring a binding" between these two elements and does not recite any details of what this "configuring" requires. Accordingly, those of ordinary skill in the art would interpret Grey's instantiation of an interface between the two to reasonably read on the claimed "configuring a binding".

5. Starting in the 1st full par. on pg. 13, the applicant states:

Claim 76 further recites:

executing the run-time operator interface application, wherein said executing comprises the control executing to automatically determine the steps in the test executive sequence, wherein the binding between the GUI element and the control causes the GUI element to automatically display at least a subset of the steps in response to the control determining the steps, wherein the GUI element displays the at least a subset of the steps in the graphical user interface of the run-time operator interface application during execution of the run-time operator interface application.

Art Unit: 2193

With respect to these limitations, the Examiner again cites the portions of Grey previously discussed above. Thus, Appellant disagrees with the Examiner for similar reasons as set forth above.

Respectfully, the applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references. Specifically, applicant's previous (and un-persuasive) arguments regarding these citations were in relation to a different limitation(s). Accordingly, the applicant's reference to the previous (and un-persuasive) arguments does not address any perceived differences between the citation and the limitations in question here.

6. Starting in the last full par. on g. 13, the applicant states:

Furthermore, Appellant also submits that the Examiner has not established a proper motivation to combine Stutz with Grey. The Examiner states:

It would have been obvious to one of ordinary skill in the art at the time the invention was made to develop the run-time operator interface disclosed in Grey using the methods taught in Stutz (col. 10, lines 46-48) because Stutz provides "an improved method ... for dynamically generating object connections (col. 8, lines 14-17).

Stutz relates generally to creating connections between objects in a program. Stutz's invention operates at a fairly low level of programming, e.g., in order to allow a source object to notify a sink object. (See Abstract). In contrast, Grey's invention operates at a much higher level and relates to a test executive system that allows a user to create a test sequence, e.g., through the graphical user interface of a sequence editor such as illustrated in FIG. 4. It is very difficult to see the particular relevance of Stutz's invention to Grey's invention or to understand how or why Stutz's teaching would be incorporated into the teaching of Grey which is cited by the Examiner. **Appellant respectfully submits that "an improved method ... for dynamically generating object connections" is an extremely vague motivation and certainly does not meet the required standard for a prima facie obviousness rejection under 35 U.S.C. 103(a).**

The examiner respectfully disagrees. First, it is not clear how the applicant is asserting Grey's controls (Fig. 4 'Main' tab, 'Step' column; col. 4, lines 4-5 "The TestStand Engine") are added to the operator interface (i.e. the GUI) if not in response to user input (e.g. a software developer building the GUI). Further, the applicant has not indicated exactly how the supposedly "extremely vague motivation" fails to meet the required standard for a prima facie obviousness rejection and thus yet again has failed to meet the requirements for persuasive arguments outlined in 37 CFR 1.111(b).

Further, Stutz discloses a means of developing an application and more particularly for configuring a binding between objects of the application (Abstract "generating object connections"). Those of ordinary skill in the art would have recognized 1) Grey's software application must be developed and 2) Stutz provides a means to develop an application. Those of ordinary skill in the art would have recognized that a means to develop an application would have been useful in developing an application. It is not at all clear why the applicant would feel such a combination would be anything less than obvious.

7. In the par. bridging pp. 14 and 15, the applicant states:

Grey teaches a test executive system which includes a built-in sequence editor and default run-time operator interfaces. The built-in sequence editor is operable to display the steps in a test sequence, e.g., as shown in FIG. 4. However, Grey does not contain any teaching regarding the user of the test executive system creating his own run-time operator interface application by including in the run-time operator interface application a control with pre-existing functionality to automatically determine and display the steps in the sequence in a GUI element, as recited in claim 76. Furthermore, Stutz does not remedy this deficiency of Grey's teaching.

The examiner respectfully disagrees. The claims do not recite "The user of the test executive system creating his own run-time operator interface application by including in the run-time operator interface application a control with pre-existing functionality to automatically determine and display the steps in the sequence in a GUI element". Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Dependent Claims 77 and 78

8. In the first full par. on pg. 16, the applicant states:

[Grey's col. 31, lines 31-33] pertains to the user defining a step type such as described above and teaches nothing whatsoever about a control that can be included in a run-time operator interface application, where the control includes pre-existing functionality for formatting the at least a subset of the steps into a formatted list.

The examiner respectfully disagrees. Grey's col. 31, lines 31-33 discuss his "General Tab". Those of ordinary skill in the art would have understood that a 'Tab Control' is a control that can be included in a run-time operator interface application (i.e. a GUI control; see e.g. Fig. 4). Further, col. 31, lines 31-33 describe pre-existing functionality of the tab used to format the step display. Specifically, the cited section discloses a name, description, comment and icon to be associated with the step and displayed in appropriate columns (see Fig. 4).

Dependent Claims 80 and 81

Art Unit: 2193

9. Starting in the last full par. on pg. 16, the applicant states:

As discussed above, [Grey's col. 31, lines 31-33] pertains to the user defining a name, description, and comment for a step type. Grey teaches nothing whatsoever about configuring a control in response to configuration user input that specifies an appearance for the steps that are displayed in the GUI element, where configuring the control enables the control to cause the steps to be displayed in the GUI element with the specified appearance.

The Examiner cites the same teaching in the rejection of claim 81. However, Grey teaches nothing regarding the specific limitations of, "wherein the configuration user input specifies one or more properties regarding a plurality of columns to display in the GUI element" and "wherein configuring the control enables the control to cause information for each displayed step to be displayed in the GUI element in the plurality of columns according to the one or more specified properties."

The examiner respectfully disagrees for the reasons discussed in conjunction with claims 77 and 78.

Dependent Claim 82

10. In the par. bridging pp. 17 and 18 the applicant states:

In the rejection of claim 82 the Examiner cites the same portions of Grey already discussed above. However, Grey does not teach the specific combination of limitations recited in claim 82, in combination with the other recited limitations of claim 82. In particular, there is no teaching regarding the bindings configured between the GUI elements and the control and the recited functionality which is caused by the bindings.

The examiner respectfully disagrees. Those of ordinary skill in the art would have recognized that the instantiation of Grey's 'interfaces' (see e.g. col. 3, lines 16-19 "the operator interface programs interface to the test executive engine") are reasonably read on the claimed bindings (interfaces) configured between the GUI elements (Fig. 4, 'main' tab; the control used to load the test file in col. 4, lines 4-5) and the control (the

Art Unit: 2193

test executive engine) and that these controls provide the recited functionality as detailed in the rejection.

11. In the 2nd to last par. on pg. 18, the applicant states:

Thus, the Examiner has referred to Grey's test executive engine in both cases. However, claim 85 recites that a control calls the test executive environment. The Examiner has equated the test executive environment with the engine itself. What then does the Examiner consider to be the control? It is not at all clear. In any case, Grey does not teach a control which calls the test executive engine (which the Examiner has equated with the recited "test executive environment") and where the control includes pre-existing first functionality for determining the steps in the test executive sequence, as recited in claims 85 and 76.

The examiner respectfully disagrees. Initially it is noted that the term "environment" is exceptionally broad and it should be clear that any system which executes a test can reasonably be read on the claimed "test executive environment". Accordingly, those of ordinary skill in the art would have recognized that Grey's interface between the control ("test executive engine") and the rest of the application (col. 3, lines 16-18 "the operator interface programs interface to the test executive engine") meets the claimed limitations.

Dependent Claim 91

12. In the 1st par. on pg. 19, the applicant states:

Dependent claim 91 is indicated as being rejected in the listing of claims in the Office Action. However, the Examiner's remarks do not address claim 91, and no rationale is given for its rejection. Appellant thus submits that claim 91 does not stand properly rejected.

The examiner acknowledges this omission and corrects it in this action.

Art Unit: 2193

Independent Claims 94 and 95

13. Starting in the 2nd to last full par. on pg. 19 the applicant states:

The Examiner rejects claim 94 for similar reasons as the rejection of claim 76, stating: "Claim 94 recites limitations similar to those addressed in the rejection of claim 76 with the exception that..." Appellant traverses the rejection of claim 94 for similar reasons as set forth above with respect to claim 76. In particular, Appellant submits that the Examiner has not established a proper motivation to combine Stutz with Grey, as argued above.

The examiner respectfully disagrees for the reasons discussed above in conjunction with claim 76.

14. In the first par. on pg. 20, the applicant states:

With respect to these limitations the Examiner cites Col. 8, lines 9-10; Col. 56, lines 48-50; and FIG. 50. The cited portions of Grey relate generally to step results being automatically collected during the execution of a test executive sequence. Grey's test executive system includes functionality for collecting and displaying the results. However, claim 94 relates to the creation of a run-time operator interface application by a user, e.g., a user of a test executive system. In particular, the user includes both a GUI element and a control in the run-time operator interface application and configures a binding between the GUI element and the control which enables the GUI element to automatically display the report in response to the control generating the report during execution of the run-time operator interface application. Thus, the user can create a run-time operator interface application which is operable to generate and display a report during the execution of the run-time operator interface application by including the GUI element and the control in the run-time operator interface application and configuring the binding between the GUI element and the control. Grey does not teach this subject matter. Furthermore, the combination of Stutz with Grey does not remedy this deficiency.

The examiner respectfully disagrees. Initially it is noted that the claim does not recite "the creation of a run-time operator interface application by ... a user of a test executive system". Further, as discussed above, Grey discloses configuring a binding

Art Unit: 2193

between various GUI elements (including the Report tab in Fig. 50) and it would have been obvious to perform this configuring according to user input.

Independent Claims 96 and 113

15. In the 2nd par. on pg. 21 the applicant states:

Appellant traverses the rejection of claim 96 for similar reasons as set forth above with respect to claim 76. In particular, Appellant submits that the Examiner has not established a proper motivation to combine Stutz with Grey, as argued above.

The examiner respectfully disagrees for the reasons discussed above in conjunction with claim 76.

16. In the par. bridging pp. 21 and 22 the applicant states:

With respect to these limitations the Examiner cites Grey at Col. 23, lines 35-39. As discussed above, Grey's test executive system includes a sequence editor with a graphical user interface that enables a user to create a test executive sequence. The cited portion of Grey teaches that, "The user can start an execution in the sequence editor by selecting the Run item or one of the process model entry points from the Execute menu." Thus, the sequence editor, which is provided to the user of the test executive system, includes a Run item that the user can select to start execution of the sequence. However, Grey does not teach the capability for a user to create his own run-time operator interface application by including both a GUI element and a control in the run-time operator interface application and configuring a binding between the GUI element and the control such that, when the run-time operator interface application is executed, the binding between the GUI element and the control causes the control to automatically invoke execution of the test executive sequence in response to the user input to the GUI element. Grey's cited teaching of the user starting execution of the test executive sequence refers to the user starting it from within the provided sequence editor and does not refer to the user creating a run-time operator interface application, and does not refer to the execution being started from within a user-created run-time operator interface application. Furthermore, the combination of Stutz with Grey does not remedy this deficiency.

Art Unit: 2193

The examiner respectfully disagrees. While it is acknowledged Grey does not explicitly disclose creating his run-time operator interface application (e.g. Abstract "A test executive program"), those of ordinary skill in the art would have recognized that it is necessary to create such an application. In other words some user of a development system created the disclosed application. Accordingly those of ordinary skill would recognize that by disclosing the application Grey inherently discloses its creation.

17. The applicant's arguments regarding claim 113 are similar to those regarding claim 96 and are likewise unpersuasive.

Dependent Claim 97

18. In the first full par. on pg. 23, the applicant states:

Claim 97 recites further limitations which are similar to those recited in independent claim 76 and discussed above. Appellant traverses the rejection of claim 97 for similar reasons as set forth above with respect to claim 76.

The examiner respectfully disagrees for the reasons discussed above in conjunction with claim 76.

Dependent Claim 98

19. In the 2nd to last full par. on pg. 23 the applicant states:

The Examiner gives no rationale for rejecting claim 98 and does not address the specific recited limitation of, "wherein said configuring the binding between the GUI element and the first control also enables the first control to invoke the second control to perform the second functionality." Appellant submits that the cited references do not teach this limitation in combination with the other recited limitations.

The examiner acknowledges this omission and corrects it in this action.

Dependent Claim 102

20. In the last par. on pg. 23, the applicant states:

Claim 102 recites similar limitations as claim 85 discussed above. Appellant respectfully submits that claim 102 is separately patentable over Grey and Stutz for reasons similar to those discussed with reference to claim 85.

The examiner respectfully disagrees for the reasons discussed above in conjunction with claim 85.

Independent Claim 114

21. The applicant's arguments regarding claim 114 (see pp. 24-35) are similar to those regarding claims 94 and 96 and are likewise unpersuasive.

Claim Rejections - 35 USC § 103

22. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

23. **Claims 76-86, 88-98, 100-107 and 109-114 are rejected under 35 U.S.C.**

103(a) as being unpatentable over US 6,401,220 to Grey et al. (Grey) in view of US 5,485,617 to Stutz et al. (Stutz).

24. **Regarding Claim 76:** Grey discloses a computer-implemented method for displaying information regarding a test executive sequence, wherein the test executive sequence includes a plurality of steps, the method comprising:

including a GUI element in a graphical user interface of a run-time operator interface application, wherein the GUI element is operable to display information (Fig. 4 see the 'Main' tab, 'Step' column);

including a control in the run-time operator interface application, wherein the control includes pre-existing first functionality for determining the steps in the test executive sequence (col. 4, lines 4-5 "The TestStand Engine automatically determines the type being loaded");

configuring a binding between the GUI element and the control (col. 3, lines 16-18 "The sequence editor ... interface[s] to the test executive engine"), wherein configuring the binding enables the GUI element to automatically display at least a subset of the steps in the test executive sequence in response to the control determining the steps in the test executive sequence during execution of the run-time operator interface application (Fig. 4 see the 'Main' tab, 'Step' column); and

executing the run-time operator interface application, wherein said executing comprises the control executing to automatically determine the steps in the test executive sequence (col. 4, lines 4-5 "The TestStand Engine automatically determines the type being loaded"), wherein the binding between the GUI element and the control causes the GUI element to automatically display at least a subset of the steps in

Art Unit: 2193

response to the control determining the steps (Fig. 4 see the 'Main' tab, 'Step' column), wherein the GUI element displays the at least a subset of the steps in the graphical user interface of the run-time operator interface application during execution of the run-time operator interface application (Fig. 4 see the 'Main' tab, 'Step' column).

25. Grey does not disclose including the GUI element and control in the run-time operator interface in response to user input.

26. Stutz teaches that including GUI elements and controls in a run-time operator interface is done in response to user input (col. 10, lines 46-48 "specifies the visual components and their location on the display").

27. It would have been obvious to one of ordinary skill in the art at the time the invention was made to develop the run-time operator interface disclosed in Grey using the methods taught in Stutz (col. 10, lines 46-48) because Stutz provides "an improved method ... for dynamically generating object connections" (col. 8, lines 14-17).

28. **Regarding Claim 77:** The rejection of claim 76 is incorporated; further Grey discloses:

wherein the control also includes pre-existing functionality for formatting the at least a subset of the steps in the test executive sequence into a formatted list (col. 31,

Art Unit: 2193

lines 31-33 “specify a name, description, and comment for the step type. The user also can specify an icon and a module adapter”);

wherein the GUI element automatically displaying the at least a subset of the steps comprises the GUI element automatically displaying the list of the at least a subset of the steps (Fig. 4 see the ‘Main’ tab, ‘Step’ column).

29. **Regarding Claim 78:** The rejection of claim 77 is incorporated; further Grey discloses:

wherein in performing said formatting the at least a subset of the steps in the test executive sequence into the formatted list, the control is operable to:

determine information regarding each of the at least a subset of the steps in the test executive sequence; and

format the information for display in the GUI element;

wherein the formatted list includes the formatted information for each of the steps in the at least a subset of the steps (Fig. 4, see ‘Main’ tab).

30. **Regarding Claim 79:** The rejection of claim 76 is incorporated; further Grey discloses:

wherein the test executive sequence is stored in a sequence file (col. 5, lines 53-54 “a test sequence file”);

wherein in automatically determining the steps in the test executive sequence, the control is operable to automatically obtain information from the sequence file

Art Unit: 2193

regarding the test executive sequence and determine the steps based on the information obtained from the sequence file (col. 4, lines 1-5 “the user loads a file ... The TestStand Engine automatically determines the type being loaded”).

31. **Regarding Claim 80:** The rejection of claim 76 is incorporated; further Grey discloses:

configuring the control in response to configuration user input after said including the control in the run-time operator interface application (col. 31, lines 31-33 “specify a name, description, and comment for the step type. The user also can specify an icon and a module adapter”), wherein the configuration user input specifies an appearance for the displayed steps, wherein configuring the control enables the control to cause the steps to be displayed in the GUI element with the specified appearance (Fig. 4, see ‘Main’ tab).

32. **Regarding Claim 81:** The rejection of claim 80 is incorporated; further Grey discloses:

wherein the configuration user input specifies one or more properties regarding a plurality of columns to display in the GUI element (col. 31, lines 31-33 “specify a name, description, and comment for the step type. The user also can specify an icon and a module adapter”); wherein configuring the control enables the control to cause information for each displayed step to be displayed in the GUI element in the plurality of columns according to the one or more specified properties (Fig. 4, see ‘Main’ tab).

33. **Regarding Claim 82:** The rejection of claim 76 is incorporated; further Grey discloses:

wherein the GUI element comprises a first GUI element;

wherein the method further comprises:

including a second GUI element in the run-time operator interface application in response to user input (col. 4, lines 4-5 “the user loading the file”; Note that those of ordinary skill in the art would have recognized this action is preformed through a GUI element.); and

configuring a binding between the second GUI element and the control (col. 3, lines 16-19 “the operator interface programs interface to the test executive engine”);

wherein executing the run-time operator interface application comprises the second GUI element receiving user input during execution of the run-time operator interface application (col. 4, lines 4-5 “the user loading the file”);

wherein the binding between the second GUI element and the control causes the control to automatically determine the steps in the test executive sequence in response to the user input received to the second GUI element during execution of the run-time operator interface application (col. 4, lines 4-5 “In response to the user loading the file, the TestStand Engine automatically determines the type being loaded”);

wherein said configuring the binding between the first GUI element and the control enables the first GUI element to automatically display the at least a subset of the

Art Unit: 2193

steps in response to the user input received to the second GUI element during execution of the run-time operator interface application (Fig. 4 see the 'Main' tab, 'Step' column).

34. **Regarding Claim 83:** The rejection of claim 76 is incorporated; further Grey does not disclose including the control in the run-time operator interface application or that configuring the binding between the GUI elements removes a need for a user to create program code for providing these functionalities.

35. Stutz teaches including a control in the run-time operator interface application enables a user to configure the run-time operator interface application to perform a first functionality without requiring the user to create program code (col. 10, lines 42-45 "a list of predefined components (objects) that can be interconnected"); and

configuring a binding between a GUI element and the control enables the user to configure the run-time operator interface application to automatically display the results of said first functionality without requiring the user to create program code for displaying the results (col. 11, lines 5-11 "Using the various commands provided by the buttons in the command area 502").

36. **Regarding Claim 84:** The rejection of claim 76 is incorporated; further Grey discloses:

wherein the test executive sequence is operable to perform one or more tests on one or more units under test (UUTs) (col. 4, lines 47-48 “A sequence comprises a series of steps, wherein a step is typically a test preformed on an instrument.”).

37. **Regarding Claim 85:** The rejection of claim 76 is incorporated; further Grey discloses:

wherein the test executive sequence is associated with a test executive environment (col. 13, lines 32-33 “The TestStand Test Executive Engine 220 is used for creating, editing, executing, and debugging sequences.”);

wherein the control is operable to call the test executive environment during execution of the run-time operator interface application to determine the steps in the test executive sequence (col. 13, lines 39-41 “The user can call the Engine API from any programming environment”).

38. **Regarding Claim 86:** The rejection of claim 76 is incorporated; further Grey discloses:

wherein the control comprises a software component constructed in accordance with an ActiveX[™] specification (col. 3, lines 30-33 “The TestStand Engine exports an ActiveX automation API”).

Regarding Claim 88: The rejection of claim 76 is incorporated; further, while not explicitly stated, it is clear from Grey’s disclosure that the control (col. 3, lines 30-33

Art Unit: 2193

“The TestStand Engine”) does not appear on the graphical user interface of the run-time operator interface application during execution of the run-time operator interface application.

39. **Regarding Claim 89:** The rejection of claim 76 is incorporated; further Grey does not disclose the control is a pre-existing control provided by an application development environment.

40. Stutz teaches a pre-existing control provided by an application development environment (col. 10, lines 42-45 “a list of predefined components (objects) that can be interconnected”).

41. **Regarding Claim 90:** The rejection of claim 89 is incorporated; further Grey does not explicitly disclose installing an application development environment on a computer system.

42. Stutz discloses both the application development environment and the control (col. 10, lines 42-45 “visual programming environment ... list of predefined components”) “implemented on a computer system” (col. 9, lines 15-21). Accordingly, both the application development environment and the control must have been installed on the computer system

Art Unit: 2193

43. **Regarding Claim 92:** The rejection of claim 76 is incorporated; further Grey does not disclose the configuring the binding between the GUI element and the control comprises performing one or more calls during execution of the run-time operator interface application.

44. Stutz discloses said configuring the binding between the GUI element and the control comprises performing one or more calls to bind the GUI element to the control during execution of the test executive application (col. 15, lines 49-52 “The function SetUpConnection ... connects the appropriate notification interface”).

45. **Regarding Claim 93:** The rejection of claim 76 is incorporated; further Grey does not disclose configuring the binding between the GUI element and the control is preformed in response to user input.

46. Stutz teaches configuring a binding between a GUI element and a control is performed in response to receiving user input to a graphical user interface to specify the binding between the GUI element and the control (col. 11, lines 5-11 “Using the various commands provided by the buttons in the command area 502”).

47. **Claim 94** recites limitations similar to those addressed in the rejection of claim 76 with the exception that the claim is directed to a control and GUI element for respectively generating and displaying a report. (see Grey col. 8, lines 9-10 “The

Art Unit: 2193

TestStand Engine operates to automatically collect the results of each step in the sequence during execution”; col. 56, lines 48-50 “TestReport ... to generate the contents of the test report”; Fig. 50, see the ‘Context’ tab, ‘ResultList’ node; also see the ‘Report’ tab).

48. **Claim 95** recites limitations similar to those addressed in the rejection of claim 76 with the exception that the claim is directed to a control and GUI element for respectively generating and displaying an execution result. (see Grey col. 8, lines 9-10 “The TestStand Engine operates to automatically collect the results of each step in the sequence during execution”; col. 56, lines 48-50 “TestReport ... to generate the contents of the test report”; Fig. 50, see the ‘Context’ tab, ‘ResultList’ node; also see the ‘Report’ tab).

49. **Claim 96-97, 99-112** recite limitations similar to those addressed in the rejections of claims 76-93 with the exception that the claims are directed to GUI element for receiving user input to cause a control to automatically invoke execution of a test executive sequence. (see Grey col. 23, lines 35-39 “start an execution ... by selecting the Run <SequenceName> item”).

50. **Regarding Claim 98:** The rejection of claim 96 is incorporated; further Stutz discloses:

wherein the control is a first control;

wherein the method further comprises including a second control in the run-time operator interface application in response to user input, wherein the second control includes pre-existing second functionality (col. 11, lines 11-15 "multiple selection list box object 509");

wherein said configuring the binding between the GUI element and the first control also enables the first control to invoke the second control to perform the second functionality (col. 11, lines 11-15 "code for updating the list of files shown in the multiple selection list box object 509").

51. **Claim 113** recites limitations similar to those addressed in the rejection of claim 96 with the exception that the claim is directed to a GUI element for receiving user input to cause a control to automatically invoke execution of a test executive sequence. (see Grey col. 24, lines 1-4 "The menus ... have commands that allow the user to stop execution").

52. **Claim 114** recites limitations similar to those addressed in the rejection of claim 76 with the exception that the claim is directed to first and second GUI elements and a control wherein the control opens a dialog box to allow a user to select a test executive sequence in response to user input received to the first GUI element (Fig. 30, see the text box labeled "File Pathname" containing the text "ComputerCPU.seq"), and wherein the second GUI element automatically displays the steps in the test executive sequence (Fig. 4 see the 'Main' tab, 'Step' column).

53. **Claim 91 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 6,401,220 to Grey et al. (Grey) in view of US 5,485,617 to Stutz et al. (Stutz) and further in view of US 6,718,534 to Carter et al. (Carter).**

54. **Regarding Claim 91:** The rejection of claim 89 is incorporated; further the Grey-Stutz combination does not disclose installing the control on the computer system after installing the application development environment.

55. Carter teaches installing a control after installing an application development environment (col. 6, lines 3-5 "The user can import a control from an external source").

56. It would have been obvious to a person of ordinary skill in the art at the time of the invention to combine the teachings of the Stutz-Grey combination in order to avoid "replication of the control programmability function" and the associated Duplication of work and increased cost. (Carter col. 1, lines 48-52).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason Mitchell whose telephone number is (571) 272-3728. The examiner can normally be reached on Monday-Thursday and alternate Fridays 7:30-5:00.

Art Unit: 2193

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Bullock Lewis can be reached on (571) 272-3759. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Jason Mitchell/
Jason Mitchell
9/2/08